IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Jenkins	**Conf. No.:**	9087
Serial No.:	10/681,908	**Art Unit:**	2193
Filing Date:	10/09/2003	**Examiner:**	Mitchell, Jason D.
Title:	COMPUTER-IMPLEMENTED METHOD, SYSTEM AND PROGRAM PRODUCT FOR DETERMINING A MESSAGE QUANTITY FOR PROGRAM CODE	**Docket No.:**	RSW920030212US1 (IBMR-0055)

Mail Stop Amendment
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 C.F.R. 1.131

I, the Applicant in the above-identified patent application, declare as follows:

1. That I am the inventor of the subject matter described and claimed in the above-identified patent application.

2. That prior to July 10, 2003, I conceived of a computer-implemented method for determining a message quantity for computer program code, comprising:

providing computer program code;

analyzing the computer program code to determine if messages are needed;

uniquely formatting sections of the computer program code based upon types of messages determined to be needed;

determining a quantity for each of the types of messages based on the uniquely formatted sections of the computer program code; and

outputting the quantity.

3. That prior to July 10, 2007, I conceived of a computerized system for determining a message quantity for computer program code, comprising:

a processor; and

a memory, the memory having:

a program analysis system for analyzing the computer program code to determine if messages are needed;

a program formatting system for uniquely formatting sections of the computer program code based upon types of messages determined to be needed by the program analysis system; and

a quantity determination system for determining a quantity for each of the types of messages based on the uniquely formatted sections of the computer program code and for outputting the quantity.

4. That prior to July 10, 2003, I conceived of a program product stored on a recordable medium for determining a message quantity for computer program code, which when executed, comprises:

program code for analyzing the computer program code to determine if messages are needed;

program code for uniquely formatting sections of the computer program code based upon types of messages determined to be needed; and program code for determining a quantity for each of the types of messages based on the uniquely formatted sections of the computer program code and for outputting the quantity.

5. That prior to November 4, 2001, we conceived of a program product stored on a recordable medium for placing logic nodes based on an estimated wiring congestion, which when executed comprising:

program code for partitioning a chip into a placement level of bins;

program code for arranging logic nodes in the bins;

program code for determining relative probabilities for potential implementations of wiring interconnects between the logic nodes that will cross over edges of the bins;

a program code for comparing, for each edge between adjacent bins, a total of corresponding relative probabilities to a wiring availability; and

program code for placing the logic nodes in the bins based on the comparison.

6. That the present invention is described in a Disclosure of Invention (Exhibit "A") submitted to the IBM Corporation Patent Department before July 10, 2003.

7. That, subsequent to the conception of the invention, and up until the patent application filing date of October 9, 2003, I diligently and actively assisted the IBM Corporation Patent Department in the planning, preparation, review, and filing of the above-identified patent application.

Declarant further states that the above statements were made with the knowledge that willful false statements and the like are punishable by fine and/or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that any such willful false statement may jeopardize the validity of this application or any patent resulting therefrom.

Date: October 17, 2007

Jana Jenkins

Jana Jenkins

Main Idea for Disclosure RSW8-2003-0344

Prepared for and/or by an IBM Attorney - IBM Confidential

Archived On [redacted]

Title of disclosure (in English)

A message methodology: A message check for code

Main Idea of disclosure

1. Background: What is the problem solved by your invention? Describe known solutions to this problem (if any). What are the drawbacks of such known solutions, or why is an additional solution required? Cite any relevant technical documents or references.

Addresses the lack of control over message creation and management. A message is considered information from a program or system that informs the user of a condition that might affect further processing
of a current program.

Anyone can create a message of any type, number, or length. Some properties files contain 10 messages, some contain over a hundred. No one knows what the correct number of messages should be and what types of messages should be created to support a certain amount of code. As a result, PII ? counts are always inaccurate and highly inflated to try to accommodate some unknown number.

My invention identifies the types of messages and the number of messages that should be created to support a particular piece of code.

2. Summary of Invention: Briefly describe the core idea of your invention (saving the details for questions #3 below). Describe the advantage(s) of using your invention instead of the known solutions described above.

Run an exec against a particular code file to identify if messages are required to support this code segment. If the need for an error, warning, or information message is detected, highlight the code in a particular color (depending on the message type), count the message types and display the results. Provide an option to save the file for future reference.

This invention identifies for the developer the number and types of messages that are needed to support a particular code file.

This invention gives the component owner, and the NLS planner an idea of the number of lines to anticipate for PII where messages are concerned. It will aid in the planning and estimating of translation costs.

3. Description: Describe how your invention works, and how it could be implemented, using text, diagrams and flow charts as appropriate.

msgcheck.PRZ

Disclosure RSW8-2003-0344

Prepared for and/or by an IBM Attorney - IBM Confidential

Created By Jana Jenkins [redacted]
Last Modified By Cathy Robbins

Required fields are marked with the asterisk (*) and must be filled in to complete the form .

***Title of disclosure (in English)**

A message methodology: A message check for code

Summary

Status		Final Decision (File)
Final Deadline		
Final Deadline Reason		
Docket Family		RSW9-2003-0212
*Processing Location		Raleigh - RSW
*Functional Area	select	(Websphere Application Server Development (McLean)) Websphere Application Server Development (McLean)
Attorney/Patent Professional		Bruce Clay/Raleigh/IBM
IDT Team	select	Steven Miller/Raleigh/IBM Art Francis/Raleigh/IBM Mark Peters/Raleigh/IBM R Redpath/Raleigh/IBM Scott Rich/Raleigh/IBM Thom Haynes/Raleigh/IBM Keith Purcell/Raleigh/IBM Virinder Batra/Raleigh/IBM Mark D Weitzel/Raleigh/IBM David Wendt/Raleigh/IBM Ajamu Wesley/Raleigh/IBM Martin Presler-Marshall/Raleigh/IBM Alan E Booth/Raleigh/IBM James Fox/Raleigh/IBM
Submitted Date		06/12/2003 05:15:18 PM EDT
*Owning Division	select	AIM
Incentive Program		
Lab		
*Technology Code		601
PVT Score		30

Inventors with a Blue Pages entry

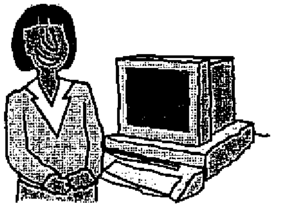
Inventors: Jana Jenkins/Raleigh/IBM

Inventor Name	Inventor Serial	Div/Dept	Inventor Phone	Manager Name
> Jenkins, Jana H.	010333	7G/AAWA	444-5359	Richey, Marva L.

> denotes primary contact

A Message Methodology

June 4, 2003

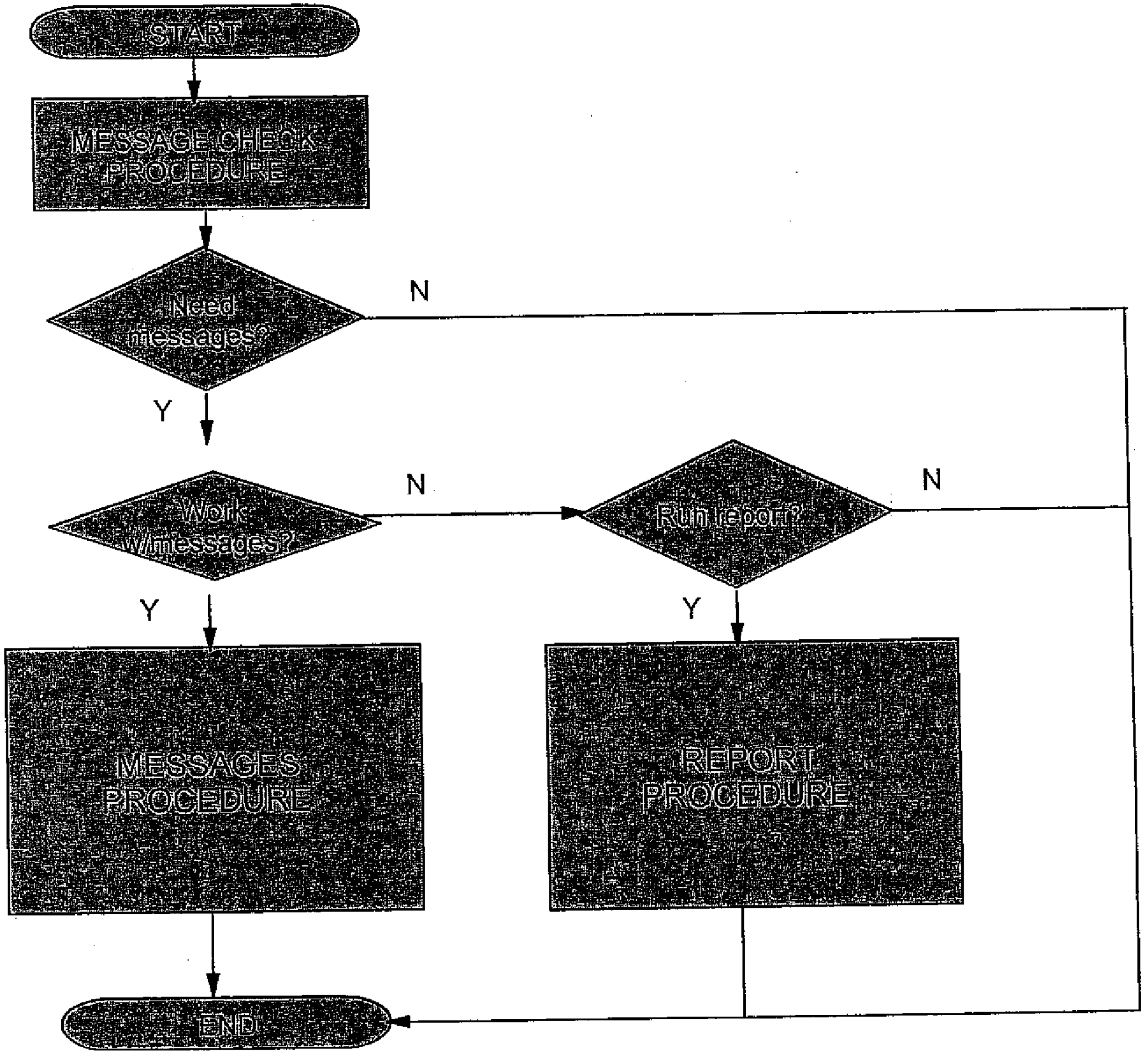


Jana Jenkins

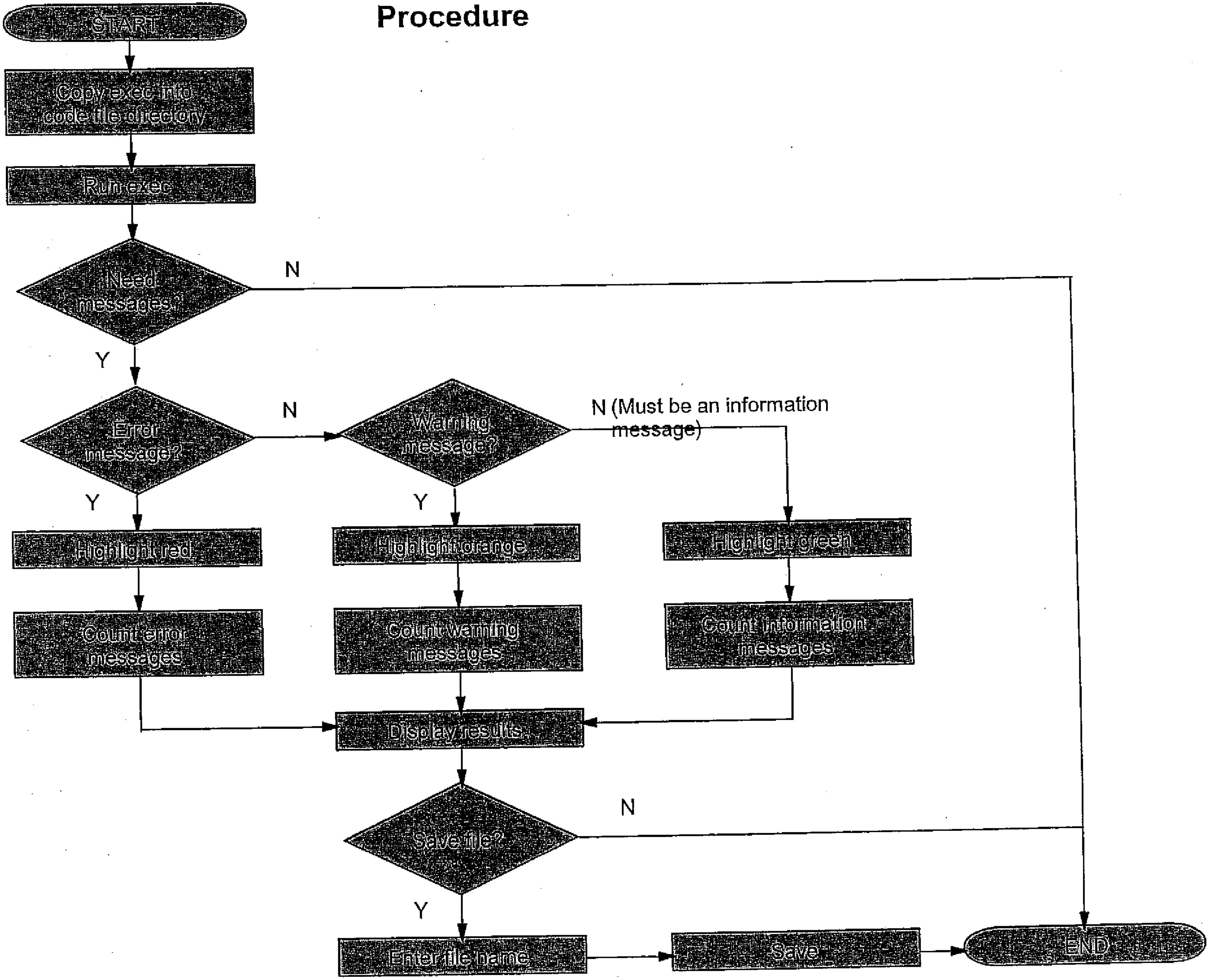
Background

- Lack of control over message creation
  - How many messages
  - What type of messages
- Need consistent way of referring to repetitive phrases
- Lack of knowledge by developers on how to create good messages
- Quality issues with current message set

Message Methodology



Message Check Procedure



Advantages to this methodology

- Provides more accurate PII count for translation
- Creates the types of messages that users need
- Creates the number of messages required to support the code
- Provides architects and code reviewers with a flag that code is too prone to error; a different approach is required
- Provides message measurements